

BlooBloop

Filippos Gleglakos, Vincent Jeannin et Théo Vuillomenet
3259.3 Développement web II
2022 – 2023 HE-Arc

Sommaire

1. Contexte
2. Conception
3. Démonstration
4. Implémentation
5. Conclusion

Contexte

Application web

Backend : Django

Frontend : Vue.js & Quasar

Déploiement : Capistrano



QUASAR



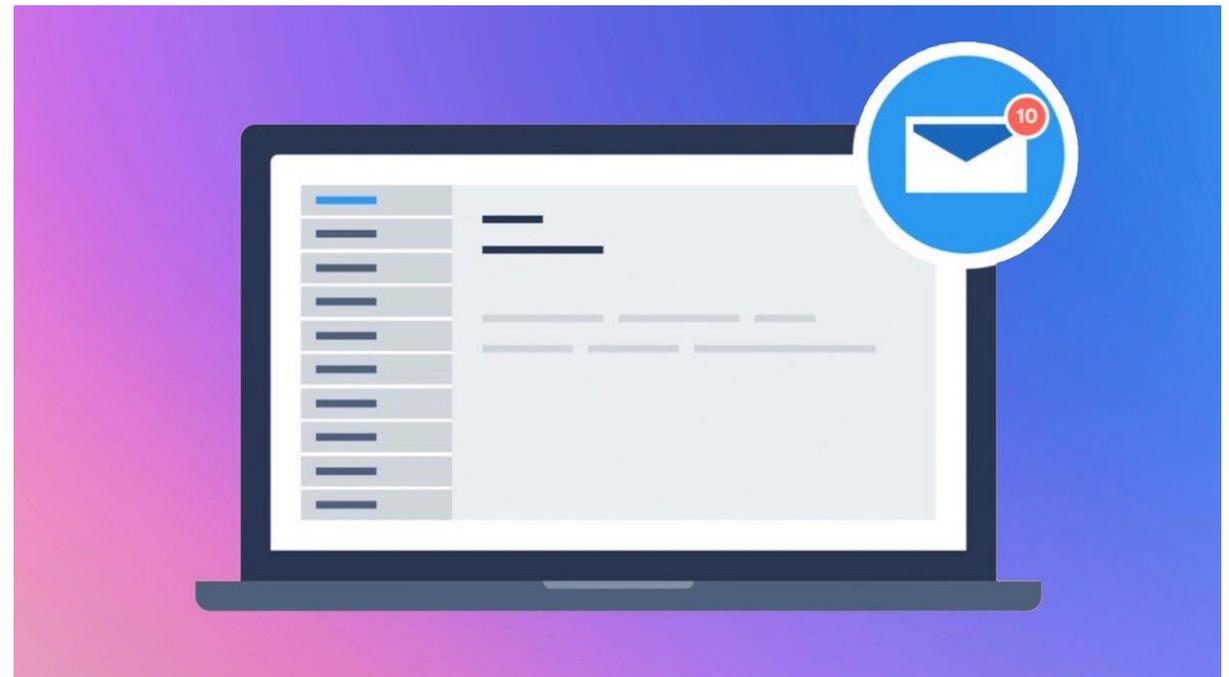
Vue.js



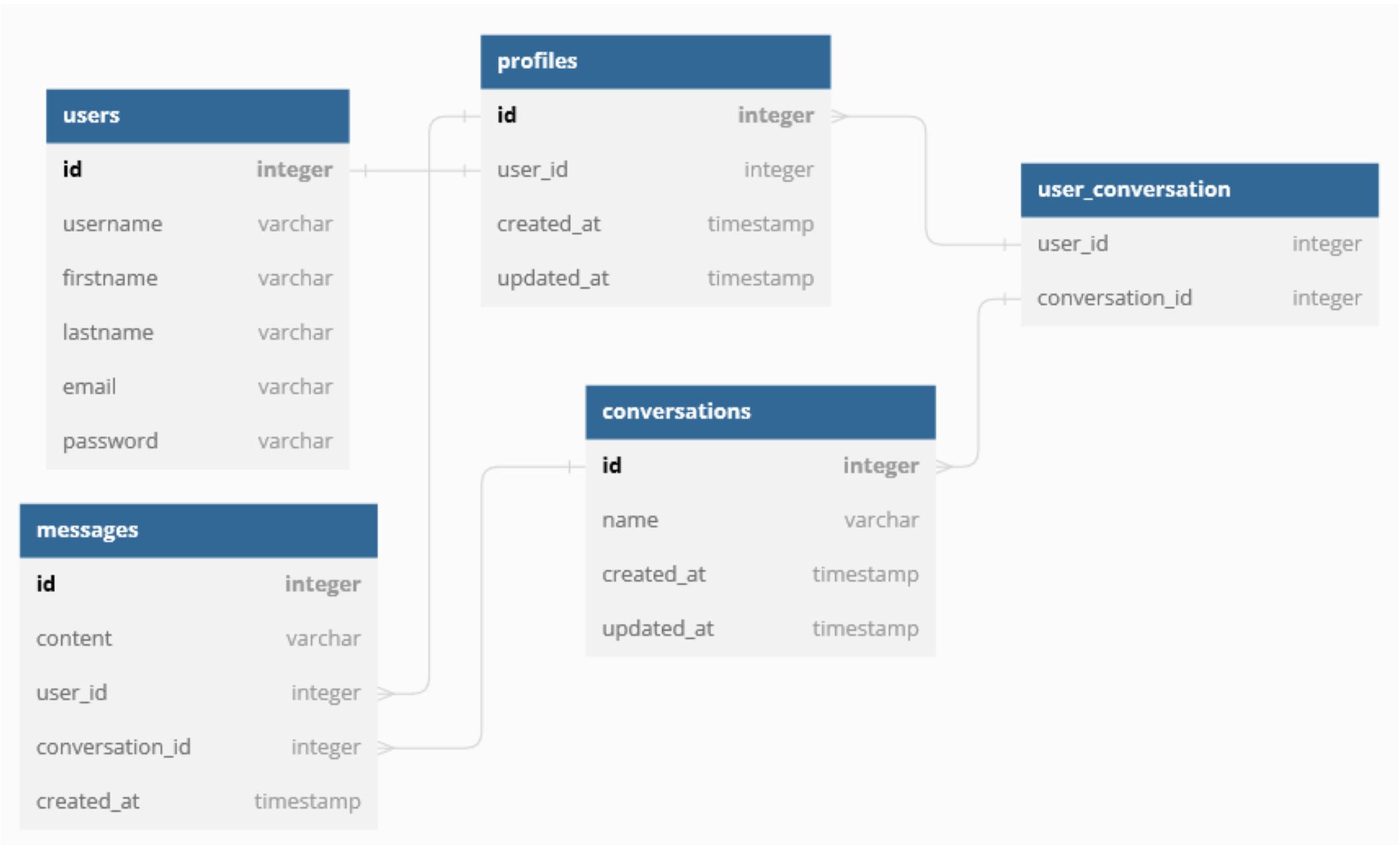
Contexte

Objectifs du projet

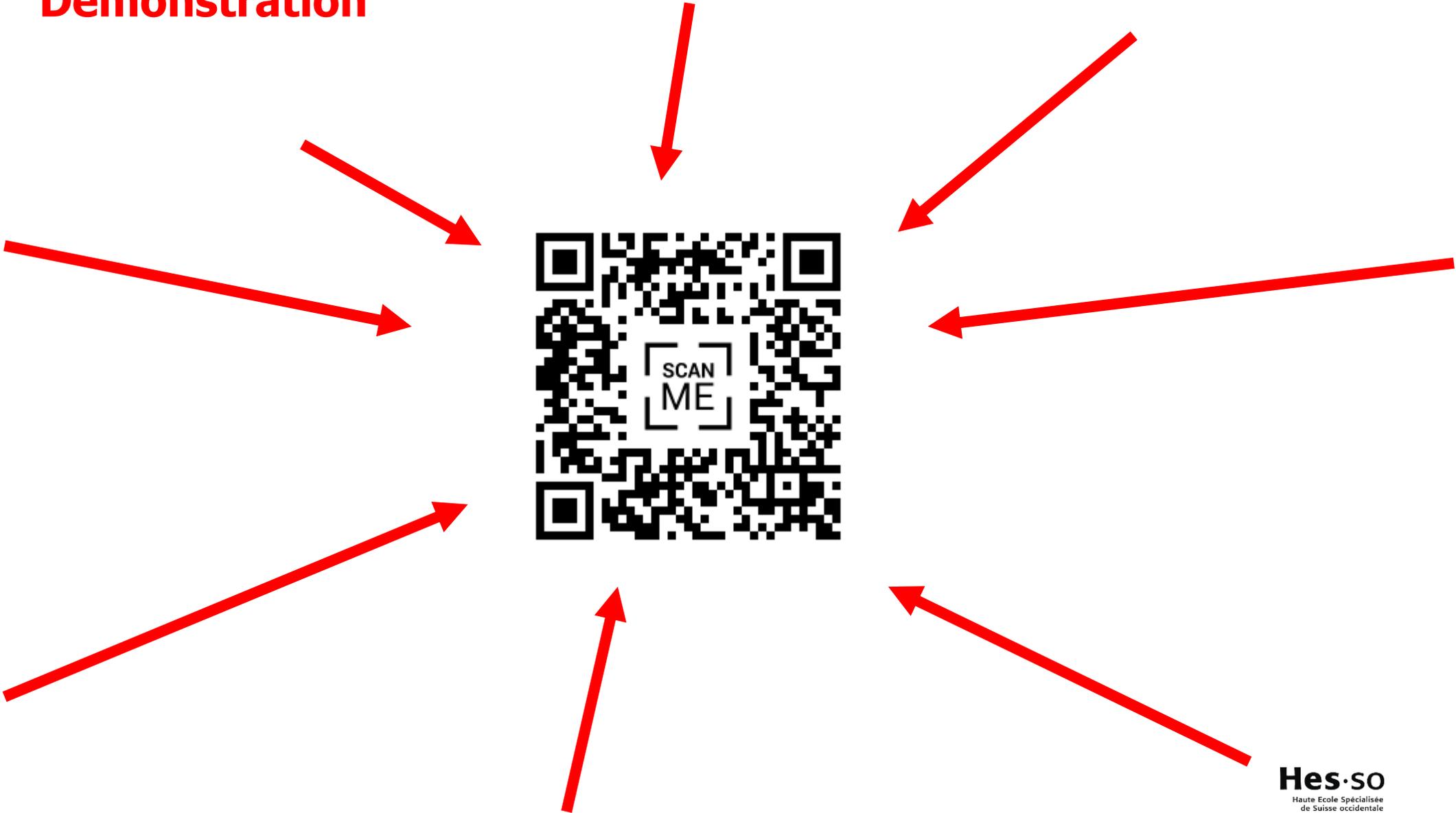
- Messagerie révolutionnaire
- CRUD Conversations
- Login
- WebSocket
- CI/CD Capistrano



Conception



Démonstration



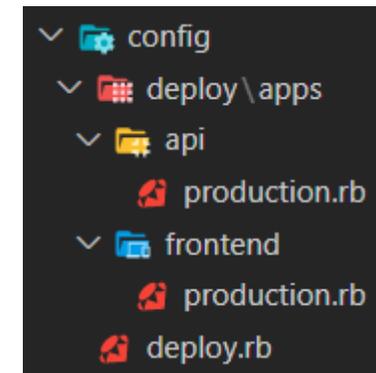
Implémentation

- CI/CD Capistrano
- Authentification
- Conversations et messages
- WebSocket



Implémentation – Capistrano

- Workshop
 - Déploiement backend
 - Action GitHub
- Adaptation : déploiement frontend
 - Double déploiement
 - Comprendre comment ça fonctionne
 - Package capistrano-multiconfig
- Unicorn restart pas



```
run: |  
  bundle exec cap apps:api:production deploy  
  bundle exec cap apps:frontend:production deploy
```

Implémentation – Authentification

- Création de user et de profil
- AxiosService
 - Intégrer automatiquement le token CSRF aux requêtes (DRY)
- Systèmes d'authentification essayés :
 - Session-based authentication
 - Token-based authentication
- Méthode pour le login
- LocalStorage pour l'affichage

Implémentation – Conversations et messages

- Implémentation du N-N
 - Faux départ
 - Actions « spéciales »
- Frontend
 - QChatMessage dans des QTabPanels
 - Filtre en fonction de l'utilisateur connecté
 - Action « spéciale »
- Son
 - Notification d'envoi et de réception
 - Classe Audio de VueJS

Implémentation – WebSocket

- Processus itératif : F5, Polling, WebSocket

```
after 'gunicorn:restart', 'gunicorn:websocket'

namespace :gunicorn do
  desc 'Launch asgi server for websocket'
  task :websocket do
    on roles(:web) do |h|
      execute :sudo, 'systemctl restart daphne.service' 5000 bloobloop.asgi:application"
    end
  end
end
```

- Différences entre local et remote
 - asgi démarré avec daphne
 - Configuration nginx
 - wss et non ws

```
proxy_set_header Host $host;
proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
proxy_set_header X-Forwarded-Host $proxy_add_x_forwarded_host;
}
```

Conclusion

- Projet axé sur des technologies (Capistrano et WebSocket)
- Objectifs atteints
- Ça fait BlooBloop !

Perspectives

- Web responsive
- Gestion utilisateur
- Réorganisation du chat
- Retravailler le son de notification

Références

Fond accueil : <https://prairiedeveloper.com/heading04/pieces-abstract-background-bubble-artik/>

Vue.js logo : <https://www.vectorlogo.zone/logos/vuejs/index.html>

Quasar logo : <https://github.com/quasarframework/quasar-template-default>

Django logo : <https://www.edgica.com/django-logo-big/>

Capistrano logo : <https://www.group-dis.com/accompagnement-devops/capistrano>

Image messagerie : <https://www.clubic.com/telecharger/actus-logiciels/dossier-367968-messagerie-10-clients-mails-pour-gerer-ses-courriers.html>

QR Code : <https://www.qr-code-generator.com/>

Icône fonctionnalité : <https://icon-library.com/icon/functionality-icon-19.html>

+ quelques captures d'écran 